AF/2124

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: **Ehnebuske et al.**

Serial No.: **09/204,973**

Filed: **December 3, 1998**

For: **Method and Apparatus for Applying Business Rules in an Object Model Driven Context**

§ Group Art Unit: **2124**
§
§ Examiner: **Ingberg, Todd D.**
§
§ Attorney Docket No.: **AT9-98-266**
§

7-10-03

Certificate of Mailing Under 37 C.F.R. § 1.8(a)
I hereby certify this correspondence is being deposited with the United States Postal Service as First Class mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on July 2, 2003.

By: Rebecca Clayton
Rebecca Clayton

35525
PATENT TRADEMARK OFFICE

TRANSMITTAL DOCUMENT

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:
ENCLOSED HEREWITH:

- Appellant's Brief (in triplicate) (37 C.F.R. 1.192); and
- Our return postcard.

A fee of $320.00 is required for filing an Appellant's Brief. Please charge this fee to IBM Deposit Account No. 09-0447. No additional fees are believed to be necessary. If, however, any additional fees are required, I authorize the Commissioner to charge these fees which may be required to Deposit Account No. 09-0447. No extension of time is believed to be necessary. If, however, an extension of time is required, the extension is requested, and I authorize the Commissioner to charge any fees for this extension to Deposit Account No. 09-0447.

Respectfully submitted,

Duke W. Yee
Registration No. 34,285
CARSTENS, YEE & CAHOON, LLP
P.O. Box 802334
Dallas, Texas 75380
(972) 367-2001
ATTORNEY FOR APPLICANTS

Docket No. AT9-98-266 *PATENT*

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: **Ehnebuske et al.** §

§ Group Art Unit: **2124**

Serial No. **09/204,973** §

§ Examiner: **Ingberg, Todd D.**

Filed: **December 3, 1998** §

§

For: **Method and Apparatus for Applying Business Rules in an Object Model Driven Context** §

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

ATTENTION: Board of Patent Appeals and Interferences

Certificate of Mailing Under 37 C.F.R. § 1.8(a)
I hereby certify this correspondence is being deposited with the United States Postal Service as First Class mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on July 2, 2003.

By: Rebecca Clayton
Rebecca Clayton

APPELLANT'S BRIEF (37 C.F.R. 1.192)

This brief is in furtherance of the Notice of Appeal, filed in this case on May 2, 2003.

The fees required under § 1.17(c), and any required petition for extension of time for filing this brief and fees therefore, are dealt with in the accompanying TRANSMITTAL OF APPEAL BRIEF.

This brief is transmitted in triplicate. (37 C.F.R. 1.192(a))

07/07/2003 EAREGAY1 00000138 090447 09204973
01 FC:1402 320.00 DA

REAL PARTIES IN INTEREST

The real party in interest in this appeal is the following party: International Business Machines, Inc.

RELATED APPEALS AND INTERFERENCES

With respect to other appeals or interferences that will directly affect, or be directly affected by, or have a bearing on the Board's decision in the pending appeal, there are no such appeals or interferences.

STATUS OF CLAIMS

A. TOTAL NUMBER OF CLAIMS IN APPLICATION

Claims in the application are: 1-28, 31-39, 41-52, 55-63, 65-76, 79-87 and 89-98

B. STATUS OF ALL THE CLAIMS IN APPLICATION

1. Claims canceled: 29, 30, 40, 53, 54, 64, 77, 78 and 88.
2. Claims withdrawn from consideration but not canceled: NONE.
3. Claims pending: 1-28, 31-39, 41-52, 55-63, 65-76, 79-87 and 89-98.
4. Claims allowed: 2-11, 48-50, 70 and 94.
5. Claims rejected: 1, 12-28, 31-39, 41-47, 51-52, 55-63, 65-69, 71-76, 79-87, 89-93 and 95-98.

C. CLAIMS ON APPEAL

The claims on appeal are: 1, 12-28, 31-39, 41-47, 51-52, 55-63, 65-69, 71-76, 79-87, 89-93 and 95-98.

STATUS OF AMENDMENTS

No amendments to the claims have been made after the Final Office Action.

SUMMARY OF INVENTION

The present invention provides a means for specifying, applying, and managing sets of temporary or permanent additions or modifications to the behavior of object-oriented programs without having to change the code of the program by using externalized rules. The points at which the externalized rules may be applied is determined by the implementation object model, thus making their specification natural to the program developers who are familiar with the program's implementation object model. In one exemplary embodiment of the present invention a pre-method control point is placed before logic of a method and a post-method control point is placed after logic of the method. A set of rules is associated with each control point based on a class of object in which the method resides, name of the method, and type of control point. The method is then invoked and as each control point is encountered during execution of the method, a determination is made as to whether the control point is active. If the control point is active, rules are selected from the set of rules associated with the control point, the rules are run, and results are obtained from running the selected rules. The selection of the rules to run may be performed using a selection algorithm. The results obtained from running the rules are then combined using a combining algorithm specified by the control point. The rules associated with the control points may perform a function that varies over time. Moreover, the rules may be associated with method context control points which are control points that may have a variety of different rules and different types of rules associated with them as the need for these rules changes over time. Furthermore, the running of the rules may affect the behavior of an object such that different rules are associated

with a control point, another control point is defined, and/or rules are associated with a second control point. In addition, the control points of a method may be activated or deactivated.

ISSUES

The only issue on appeal is whether all of claims 1, 12-28, 31-39, 41-47, 51-52, 55-63, 65-69, 71-76, 79-87, 89-93 and 95-98 are anticipated by Martin, "Principles of Object-Oriented Analysis and Design."

GROUPING OF CLAIMS

The claims do not stand or fall together. The claims stand or fall in accordance with the following groupings of claims, the reasons for these groupings being provided in the arguments presented hereafter:

Group I:	claim 1;
Group II:	claims 12, 13, 15, 23-28, 35, 46, 51, 52, 59, 72, 74-76, 83, 96 and 98;
Group III:	claim 14;
Group IV:	claims 31, 55 and 79;
Group V:	claims 32, 43, 45, 56, 67, 69, 80, 91 and 93;
Group VI:	claims 33, 44, 57, 68, 81 and 92;
Group VII:	claims 34, 58 and 82;
Group VIII:	claim 16;
Group IX:	claims 17 and 22;
Group X:	claim 18;
Group XI:	claim 19;
Group XII:	claim 20;
Group XIII:	claim 21;
Group XIV:	claims 23 and 27;
Group XV:	claims 36-39, 41, 60-63, 65, 84-87 and 89;
Group XVI:	claims 42, 47, 66, 71, 90 and 95; and

Group XVII: claims 73 and 97.

ARGUMENT

I. Examiner's Interpretation

It is first necessary to address the alleged "interpretations" made by the Examiner of the terms "modeling" and "Flow Control." The Examiner's "interpretations" of these terms in the "Examiner's Interpretation" section of the Final Office Action in no way limits Applicants' claimed invention. The terms in the claims must be interpreted in light of the specification as one of ordinary skill in the art would interpret these terms, not the use of such terms in the reference the Examiner intends to use to reject the claims nor the Examiner's own personal belief as to what the term means.

To the contrary, the Examiner in supposedly "interpreting" the term "modeling" only refers to sections of the Martin reference, discussed hereafter with regard to the rejection under 35 U.S.C. § 102. The Examiner does not interpret the term "modeling" in light of the present specification. Thus, the claims are not bound by the Examiner's alleged "interpretation." Moreover, the Examiner has merely made a general allegation as to what the term "modeling" is believed to be in view of the Martin reference and has not established in any way how such an "interpretation" has any bearing on any particular feature of any of the rejected claims. Thus, the Examiner's "interpretation" should not be regarded as limiting the scope of the present claims to what the Martin reference teaches, but rather to what the claimed features recite in view of the present specification.

Furthermore, with regard to the "interpretation" of the term "Flow Control", the Examiner admits that Applicants do not claim flow control. Thus, the claims cannot be limited by the Examiner's "interpretation" of the term "Flow Control." With regard to whether what is recited in the claims may be "a product" of "flow control," this is irrelevant to whether the claims as a whole are directed to patentable subject matter. Flow control is not claimed and thus, the claims should not be limited to the Examiner's interpretation of the meaning of "flow control". The mere use of the phrase "method logic is continuous" in any of the claims does not suddenly invoke an interpretation of "flow control" such as that alleged by the Examiner in any manner

that would limit the scope of those claims.

With regard to the Examiner's "interpretation" of the term "decorating," the present specification on page 4 clearly describes a mechanism by which a decorator pattern is used to add a new behavior to an object to thereby generate a "decorated" object. Thus, the term "decorating" in claim 3 should be interpreted in light of the specification, not the personal interpretations of the Examiner.

These arguments were first presented to the Examiner in Appellants' Response filed January 30, 2002. In response to Appellants' arguments, the Examiner in the Final Office Action alleges that these terms are terms used by artisans of ordinary skill in the art. With regard to the term "modeling" the Examiner alleges that Appellants' are unable to "distinguish the term from the specification." It is not necessary for Appellants to "distinguish the term from the specification" since the Examiner has not shown how the interpretation of the term "modeling" has any bearing on the claimed invention. The only place that any form of the term "modeling" is used in the rejected claims is in claim 13 which recites "wherein the rules perform a variety of actions conditioned by the fact that rules may be associated with particular, regularly occurring points in the object model" (emphasis added). Therefore, the interpretation of the term "modeling" is irrelevant to all of the other claims since none of these claims recite "modeling." These other claims include claim 12 from which claim 13 depends and thus, claim 12 contains features that are not limited by the "interpretation" of the term "modeling" alleged by the Examiner.

With regard to the term "flow control" the Examiner responds to Appellants' arguments by stating that Appellants are matching terms and not concepts. Appellants are not merely matching terms. To the contrary, Appellants are alleging that "flow control," as the Examiner interprets this term, has nothing to do with the claimed invention and thus, the Examiner is alleging interpretations of terms that have no bearing on the scope of the claims. Appellants are only requesting that the claims be interpreted based on the features recited therein in view of the specification – not some other unclaimed and unrelated term and "concept" the Examiner wish to "interpret" the claims in view of. However, the Examiner continues to insist on attempting to bring in unclaimed "concepts" in an attempt to try and provide an appearance that the cited reference, Martin, teaches more than it actually does as well as read in limitations to the claims that are not there. Appellants' request that the Board clearly state the improper nature of the

Examiner's actions in this regard and that the present claims are not to be limited by the Examiner's irrelevant interpretations of terms and "concepts" discussed above.

In response to Appellants' arguments with regard to the term "decorating", the Examiner states that Appellants have been "provided another opportunity to make a distinction but apparently is unable to do so." In Appellants' response to the Examiner's interpretation of the term "decorating" Appellants pointed to page 4 of the specification as teaching a mechanism for decorating an object and asserted that this term in the claims should be interpreted in light of the specification. Thus, Appellants have offered a source for interpreting this term in Appellants' own disclosure, however the Examiner continues to try and insert limitations into the claims by "interpreting" this term in his own way without regard to what Appellants' own disclosure describes. This is clearly improper and Appellants again request that the Board indicate that the Examiner's interpretation of the term "decorating" not be allowed to be limiting on the present claims.

In summary, none of the Examiner's personal interpretation or alleged interpretations in view of the Martin reference with regard to the above terms may be used to limit the scope of the pending claims for the reasons noted above. To the contrary, the terms in the claims should be interpreted in light of Appellants' specification and not the references cited against the application or the Examiner's own personal beliefs.

II. 35 U.S.C. § 102, Anticipation

The Office Action rejects claims 1-98[1] under 35 U.S.C. § 102(a) as being anticipated by Martin, "Principles of Object-Oriented Analysis and Design," published June 1, 1992. This rejection is respectfully traversed.

[1] It is noted that only claims 1, 12-28, 31-39, 41-47, 51-52, 55-63, 65-69, 71-76, 79-87, 89-93 and 95-98 are rejected since claims 29, 30, 40, 53, 54, 64, 77, 78 and 88 have been canceled and claims 2-11, 48-50, 70 and 94 have been allowed.

A. Group I - Claim 1

With regard to claim 1, the Office Action states:

> Martin anticipates a computer implemented process for applying a set of rules (Martin, Chapter 10, RULES, and page 138-139 and 249-251), the process comprising:
>
> (a) placing a pre-method control before logic of a method (Martin, page 142, operation precondition) and post method control point after the logic of the method (Martin, page 142, post condition)
>
> (b) associating a set rules with each control point (Martin, page 142, 147 "Operation" as per (a) above) based on a class of object in which the method resides (Martin, page 143, "...rules associated with diagrams of OO..."), name of the method and type of control point, whether the pre-method control point or the post-method control point (Martin, page 142, operation precondition);
>
> (c) invoking the method (Martin, page 116), wherein encountering each control point during the execution of the method comprises (Martin, page 142, post condition)
>
> (i) determining if the encountered control point is active (Martin, page 122, IF structure in center diagram);
>
> (ii) on the basis of an active control point (Interpreted as the result of the IF structure above further described in Appendix A on page 381 Control Conditions);
>
> 1) selecting rules based on a set of rules associated with the active control point associated in step (Martin, page 122, first diagram example is the control condition to fire missiles)(b);
>
> 2) running the selected rules (Martin, page 122, rule that lead to the control condition);
>
> 3) obtaining results from running the rules (Martin, page 122, trigger rule at the bottom of the page); and
>
> 4) combining the results using a combining algorithm specified by the control point (Martin, page 122, A control condition can function as a combining algorithm as seen in diagram in middle of the page and page 126 Figure 9.9 and Martin teaches a way to have a combining algorithm where one of three operations are selected as on page 124, and Martin teaches a way to have a combining algorithm where on can be selected as taught in the mutually exclusive notation on the bottom of the page 125).

A prior art reference anticipates the claimed invention under 35 U.S.C. § 102 only if every element of a claimed invention is identically shown in that single reference, arranged as they are in the claims. *In re Bond*, 910 F.2d 831, 832, 15 U.S.P.Q.2d 1566, 1567 (Fed. Cir. 1990). All limitations of the claimed invention must be considered when determining patentability. *In re Lowry*, 32 F.3d 1579, 1582, 32 U.S.P.Q.2d 1031, 1034 (Fed. Cir. 1994). Anticipation focuses on

whether a claim reads on the product or process a prior art reference discloses, not on what the reference broadly teaches. *Kalman v. Kimberly-Clark Corp.*, 713 F.2d 760, 218 U.S.P.Q. 781 (Fed. Cir. 1983). Applicants respectfully submit that Martin does not identically show each and every feature of the pending claims arranged as they are in the claims. Furthermore, Applicants respectfully submit that the claims do not read on the processes specifically described in the Martin reference and that the Office Action is engaged in applying broad "conceptual" teachings of Martin without regard for the actual specific teachings of Martin or the specific features recited in the claims.

Chapters 9 and 10 of Martin, which are the basis for the Office Action's rejection of all of claims 1-98, teach a method of modeling the behavior of an object oriented system. The modeling involves representing operations, preconditions of the operations, post conditions of the operations, control conditions, events and triggers. In the modeling described by Martin, an operation may have a precondition that identifies what must happen before the operation executes, and a post condition that describes the result of the operation if the operation executes with the precondition being satisfied. The operation may further include a control condition which is a condition that is used to determine whether the operation is to execute or not. The control condition must be checked prior to invoking the operation and may be a complex collection of Boolean conditions (page 122). Martin further teaches the implementation of these preconditions, post conditions and control conditions as rules in Chapter 10.

Martin does not teach "associating a set of rules with each control point based on a class of object in which the method resides, name of the method and type of control point," "determining if the encountered control point is active," and "selecting rules based on a set of rules associated with the active control point associated in step (b)," (emphasis added) as recited in claim 1.

With regard to the feature of "associating a set of rules with each control point based on a class of object in which the method resides, name of the method and type of control point," the Office Action alleges that this feature is taught by Martin on pages 142, 143, and 147. However, these sections of Martin have nothing to do with associating a set of rules with a control point based on a class of object in which the method resides, name of the method and type of control point. Page 142 merely describes that the event diagram is an executable diagram from which program code may be generated using a tool such as the OO-CASE tool. Page 143 merely

describes that rules may be either object state rules or object behavior rules; that rules may be used with other types of diagrams other than event diagrams; and that rules may be stated as English expressions which may then be used to generate code. Page 147 merely shows examples of an operation, event diagram and state transition diagram that have rules attached to them.

There is nothing in these sections, or any other sections, of Martin that can remotely be considered to teach the feature of "associating a set of rules with each control point based on a class of object in which the method resides, name of the method and type of control point" (emphasis added). It is not clear where or how the Office Action can extract any teaching from Martin that even suggests the features of the presently claimed invention. Rather, it appears that the Office Action is engaged in taking broad teachings of event diagrams and generating code from event diagrams, and reading into these broad teachings the very specific features recited in Appellants' claims.

However, the Office Action cannot read into the prior art features that are only present in Applicants' own disclosure and use the "modified" prior art to support a rejection of the claims. This modification of the prior art being made by the Examiner in the Final Office Action is more akin to a rejection under 35 U.S.C. § 103(a) than the actual basis of the rejection which is 35 U.S.C. § 102(a). As was established for obviousness rejections, an applicant's teachings may not be read into the prior art. *Panduit Corp. v. Denison Mfg. Co.*, 810 F.2d 1561, 1575 n. 29, 1 U.S.P.Q. 1593, 1602 n. 29 (Fed. Cir. 1987). While the court is addressing an obviousness determination in *Panduit*, the same principle applies even moreso to a rejection under 35 U.S.C. § 102 rejection since under § 102 the rejection must be based solely on the teachings of the references themselves. In this case, there is nothing in the Martin reference that can be used to support a position of anticipation with regard to the above feature.

Similarly, there is nothing in the Martin reference that remotely even hints at the feature of "determining if the encountered control point is active." The Martin reference does not even recognize a possibility of having active or inactive control points. The Office Action equates the operation precondition and post conditions to the control points recited in the claims, even though they are not the same as the recited control points. A control point, as defined in the present specification is a point at which rules may be attached to add additional functionality. The preconditions and post conditions of Martin are merely requirements for the operation to execute properly.

However, assuming that the preconditions and post conditions are the same as a control point, *arguendo*, Martin describes the precondition and post conditions as always having to be satisfied in order for proper operation of the object oriented system model. Thus, the precondition and post conditions must always "active" in the Martin reference and there is no need to determine if they are active or inactive. It is for this reason that Martin does not mention anywhere in the entire reference, any step of determining if a control point is active.

The Office Action alleges that this feature is taught in Martin at page 122 simply because Martin teaches an IF structure. The Office Action equates the precondition with the control point and then states that the presence of an IF structure in the precondition is the same as determining if a control point is active. This does not make any sense since, as is clearly described in the present invention and recited in the claims, if a control point is not active, the rules of the control point are not executed. If the precondition of Martin were the same as a control point, then the precondition must always be active in order for the IF structure to even operate. Therefore, the IF structure would always indicate that the precondition is active and thus, there is no need for the IF structure. Thus, the IF structure is not the same as the step of determining if a control point is active.

The Examiner is attempting to generalize the claimed invention and generalize the teachings of the reference in an attempt to reject the generalization of the claim as being "anticipated" by the generalization of the reference. Claim 1 clearly recites "determining if the encountered control point is active." There is no such similar step in Martin. The "IF" structure referenced by the Examiner is a precondition that is used to determine if a requisite condition exists for the following operation to be performed. There is no teaching in Martin regarding turning this condition on or off, i.e. making it active or inactive. If the precondition exists, it is required for the operation to be performed.

Just as with the above, Martin also provides no teaching that is remotely similar to the feature of "selecting rules based on a set of rules associated with the active control point associated in step (b)." While Martin teaches that rules may be used to implement the precondition and post condition of an operation, there is no teaching in Martin of associating a set of rules with an active control point and then selecting rules based on this set of rules. Martin provides no teaching at all regarding selection of rules and the Office Action has not pointed out with particularity any section of Martin that teaches a selection of rules based on a set of rules

associated with the active control point.

The Office Action alleges that this feature is taught by Martin in the first diagram on page 122 (although the following text indicates that the Examiner is in actuality referring to the last figure on page 121). These diagrams merely illustrate the use of control conditions that are checked prior to an operation executing. There is no selection of rules even mentioned or shown in these figures, let alone the selection of rules based on a set of rules associated with an active control point. That is, using the middle figure of page 122 as exemplary, there is no teaching in Martin to select the second AND clause from the IF statement in the figure as opposed to selecting the third AND clause. All of the conditions set forth in the IF clause in the figure must be satisfied in order for the operation to execute. There is no selection of rules from a set of rules associated with a control point upon determining that the control point is active during execution of a method, as recited in claim 1. The Office Action is referencing portions of the Martin reference that do not even have anything to do with the features of the claim.

Furthermore, the figures referred to in the Office Action cannot be interpreted in any way to teach the selection of rules based on a set of rules associated with an active control point. The figure illustrating firing of a missile is provided to show how a control point may be used to handle a plurality of triggers that are required for the operation to execute. The first diagram on page 122 also illustrates this concept. However, nowhere in the figures of the accompanying text is there anything mentioned about selection of rules based on a set of rules associated with an active control point.

On page 122 Martin does state that the control condition may be a collection of "or" conditions applied to triggers (see the middle diagram on page 122). However, the "or" Boolean condition does not constitute a selection of rules based on a set of rules associated with an active control point. Rather, the "or" conditions merely operate to state that if any one of the triggers satisfies the control condition, then the operation will execute. There is no selection of rules.

Thus, Martin does not teach each and every feature of claim 1 as is required under 35 U.S.C. § 102(b). Accordingly, Applicants respectfully request withdrawal of the rejection of claim 1 under 35 U.S.C. § 102(b).

Examiner's Response to Appellants' Arguments Regarding Group I

In response to Appellants' arguments, with regard to the feature of "associating a set of rules with each control point based on a class of object in which the method resides, name of the method and type of control point," the Examiner states that "the reference explicitly states the diagrams generate code not "might" as the Applicant has stated" (Final Office Action, page 35) and that Appellants' arguments amount to a general allegation that the claims define a patentable invention without specifically pointing out how the language of the claims patentably distinguishes them from the references (Final Office Action, page 36). Appellants respectfully disagree.

Appellants have set forth in their arguments the specific claim language that is not taught in the reference. Appellants has stated that the portion of the Martin reference cited by the Examiner teaches that the event diagram is an executable diagram from which program code may be generated using a tool such as the OO-CASE tool (Page 142), that rules may be either object state rules or object behavior rules; that rules may be used with other types of diagrams other than event diagrams; and that rules may be stated as English expressions which may then be used to generate code (Page 143), and examples of an operation, event diagram and state transition diagram that have rules attached to them (Page 147). Appellants have asserted that there is nothing in these cited sections of the Martin reference that teach a set of rules being associated with each control point "based on a class of object in which the method resides, name of the method and type of control point" (emphasis added). The Martin reference does not teach anything like this feature as set forth in the above arguments. Thus, Appellants' argument clearly satisfy the requirement of 37 CFR 1.111(b) with regard to "specifically pointing out how the language of the claims patentably distinguishes them from the references", i.e. there is nothing in Martin that is remotely similar to a set of rules being associated with each control point "based on a class of object in which the method resides, name of the method and type of control point."

With regard to the determining if an encountered control point is active, the Examiner states that the arguments do not satisfy 37 CFR 1.111(b) and that the term "active" is being interpreted as a factor of flow control and that this is inherent in programming and "not able to be separated out." Appellants respectfully submit that Appellants have stated the features of the claims that are not taught by the reference, what the reference actually teaches at the cited

portions, and asserted that these teachings are not sufficient to anticipate the claimed feature. Thus, Appellants have satisfied the requirements of 37 CFR 1.111(b).

In addition, there is nothing in the general concept of "flow control" that teaches to determine if a control point that is encountered during the execution of a method is active. The Examiner alleges in his "interpretation" of "flow control" that flow control is the tracing of the path of an executing of a program and that the exact path the program follows is determined by the values of the attributes and the control conditions encountered (Final Office Action, page 3). Even if what the Examiner alleges (without any support from any cited references) is true, this has no bearing on the specific features recited in claim 1. Even if flow control generally teaches that paths of execution are determined based on values of attributes and control conditions, this in no way anticipates the specific feature of determining if a control point is active when it is encountered during execution of a method, as recited in claim 1. In fact, this alleged general teaching does not even mention control points, that they can be active or inactive, that they are encountered during the execution of a method, or that when they are encountered during execution of a method that they are checked to determine if the encountered control point is active.

Moreover, there is nothing in the cited reference, even assuming that the precondition and post-conditions of Martin are equivalent to a pre-method control point and a post-method control point, which they are not, that teaches checking to see if the precondition is active or not. Again, the Examiner is generalizing the alleged teaching of "flow control", generalizing the specific features of the claimed invention, and then alleging that the claimed invention is anticipated because the two generalization are equivalent. This is clearly improper and does not meet the requirements of establishing anticipation under 35 U.S.C. § 102.

Additionally, the Examiner states that this feature is anticipated because "the Martin reference teaches a plurality of programming constructs," alleges that Appellants are limiting the use of the constructs and alleges that the reference does not put limits on the constructs. In other words, the Examiner is admitting that the Martin reference only teaches generalities while the present claims are reciting specific features. Appellants respectfully submit that this is the very problem with the Examiner's rejection of the claims, i.e. the Examiner is not examining the specific features recited in the claims but has generalize both the claimed invention and the Martin reference to a point where he believes he can allege that the reference anticipates the

claimed invention. This is clearly improper because the Examiner is not taking into account each and every feature of the invention as recited in the claims. Moreover, this statement is practically an admission by the Examiner that he has not shown where each and every feature of the claimed invention is "identically shown" in the Martin reference "arranged as they are in the claimed invention" as is required under 35 U.S.C. § 102 but instead has shown alleged constructs without limitations.

With regard to Appellants' argument regarding the feature of "selecting rules based on a set of rules associated with the active control point associated in step (b)," the Examiner responds by stating that the disagreement appears to be based on the term "active" which the Examiner interprets as a factor of flow control. The issue with regard to "flow control" has been address above. Moreover, the "disagreement" is not based on the term "active" but is rather based on the fact that the Martin reference does not teach a set of rules being associated each control point based on a class of object, a name of the method, and a type of control point, does not teach determining if an encountered control point is active, and thus, does not teach selecting rules based on the set of rules associated with the active control point on the basis of the control point being determined to be active. Not so much as one of the above features have been shown to be present in the Martin reference, whether or not the term "active" can be interpreted as being part of "flow control." Thus, the Examiner has not established anticipation with regard to claim 1.

In view of the above, Appellants respectfully submit that claim 1 is not anticipated by Martin. Appellants respectfully request withdrawal of the rejection of claim 1 under 35 U.S.C. § 102(a).

B. Group II - Claims 12, 13, 15, 23-28, 35, 46, 51, 52, 59, 72, 74-76, 83, 96 and 98

With regard to claims 12, 23, 27, 46, 51, 72, 75 and 96, Martin does not teach a method context control point as recited in each of these claims and as discussed above. A method context control point is a control point that may have a variety of different rules and different types of rules associated with it as the need for these rules changes over time (see page 11, lines 17-29 of the present specification). Martin does not teach such a method context control points. Rather, Martin teaches preconditions, post conditions, and control conditions that perform specific functions that are not intended to change over time and are not intended to have different

rules and different types of rules associated with them. This is because these conditions are linked to the operation so closely that the conditions cannot be changed without causing the operation to not execute properly.

The method context control point of the present invention allows the object model to be modified without having to change the underlying model itself. By applying new rules at active control points, new functionality is available without having to modify the model. With the event diagrams of Martin, the model itself must be modified, i.e. the operations and their preconditions, post conditions and control conditions, in order to obtain new functionality. Thus, the present claims do not read on the Martin reference and the Martin reference does not anticipate the presently claimed invention.

Therefore, for example, Martin does not teach "encountering an active control point during the invocation of the method, wherein the active control point is an active method context control point" or "selecting rules associated with the method of the object at the control point" as recited in claim 12. Moreover, Martin does not teach "encountering a method context control point" or "finding at least one rule associated with the method context control point" as recited in claim 46. Likewise, Martin does not teach "defining means for defining at least one control point, wherein the at least one control point is a method context control point" as recited in claim 51. Similar recitations of the "method context control point" are found in claims 72, 75 and 96 (as clearly illustrated in the attached appendix).

With regard to claims 23 and 27, Martin does not teach a method context control point, as discussed above. Specifically, claim 23 recites defining first and second control points of a method, the first and second control points also being a method context control point. Claim 27 recites defining at least one control point, wherein the at least one control point is a method context control point and placing this method context control point in a method after the method logic of the method. Martin does not teach these features of claims 23 and 27.

Examiner's Response to Appellants' Arguments Regarding Group II

The Examiner stated on page 42 of the Final Office Action that Appellants' arguments with regard to the "method context control point" offered with regard to claim 2 were persuasive. Thus, it appears that the Examiner is in agreement that the Martin reference does not teach

method context control points and as a result, has allowed claims 2, 48, 70 and 94. However, the Examiner alleges that the feature of a "method context control point" is not recited in claims 12, 46, 51, 72, 75 and 96 (Final Office Action, page 45). The Examiner also alleges that claims 23 and 27 do not have the features for which the allowed claims were allowed (Final Office Action, page 50-51). Appellants have shown above that each of these claims contain recitations to a "method context control point" and thus, Martin, not having taught such a feature, does not anticipate these claims.

Moreover, the only response to Appellants' arguments presented with regard to the feature of a "method context control point" is that some of the claims do not contain the allowable feature of claim 2. This in no way rebuts any of the arguments presented by Appellants with regard to the claims reciting a "method context control point" and Martin not teaching such a structure. Simply alleging that the claims do not include an allowable feature of another claim does not illustrate how Martin teaches a method context control point. There simply is no structure in Martin that provides a control point that may have a variety of different rules and different types of rules associated with it as the need for these rules changes over time. Thus, the Examiner has not established where such a feature is taught in Martin and has not established anticipation with regard to any of these claims.

Since Martin does not teach method context control points, Martin does not teach each and every feature recited in claims 12, 23, 27, 46, 51, 72, 75 and 96 as is required under 35 U.S.C. § 102(a). At least by virtue of their dependency on claims 12, 23, 27, 46, 51, 72, 75 and 96, Martin also does not teach or suggest the features recited in dependent claims 13-15, 24-26, 28, 31-35, 52-59, 73-74, 76, 79-83, and 97-98. Accordingly, Applicants respectfully request withdrawal of the rejection of claims 12-15, 23-28, 31-35, 46, 51-59, 72-76, 79-83 and 96-98 under 35 U.S.C. § 102.

C. Group III – Claim 14

In addition to the above, Martin also does not teach many of the features recited in the dependent claims set forth above. For example, Martin does not teach that the rules perform at least one function that varies over time, as recited in claim 14. As mentioned above, the precondition, post condition and control condition are only used to determine whether an

operation is to execute and whether an operation has executed correctly. There is nothing in Martin that teaches a control point that has rules that perform a function that varies over time. None of the functions of the precondition, post condition or control condition in Martin vary over time. Rather, they are fixed conditions that must be present for the associated operation to be performed. There is no change in the condition that must be present.

The Office Action alleges that this feature is taught by Martin at page 117 simply because Martin teaches clock events and that Martin teaches associating rules with event diagrams. The clock events have nothing to do with preconditions, post conditions and control conditions. The clock events are separate entities in the event diagram from preconditions, post conditions and control conditions. The Office Action is again taking general teachings and conjuring up alleged specific teachings with no actual support in the reference. The only place that control points having rules that perform functions that vary over time are even mentioned is in Applicants' own disclosure. While clock events are triggered based on time, this in no way teaches a control point that has a rule that performs a function which varies over time.

Examiner's Response to Appellants' Arguments Regarding Group III

On page 46 of the Final Office Action, the Examiner merely responds to the above arguments with regard to Group III that "the clock event programming construct in Martin is not limited." This does not address the specific arguments presented by Appellants. That is, just because the clock event may not be "limited", *arguendo*, does not mean that it is inherent in the teachings of Martin that the clock event is a precondition or post-condition that varies over time or that the precondition and post-condition are the same as the claimed pre-method control point or post-method control point. Again, the Examiner appears to think that any generalization anticipates each and every possible claim feature that may be generalized to what the Examiner believes is an equivalent structure. Appellants are not claiming a generalization of claim 14, but rather the specific invention recited in claim 14. The Examiner cannot merely disregard the specific features of the claim by refusing to examine the claimed features and examining some generalization instead. In this case, the Examiner has not shown where each and every specific feature of claim 14 is taught in Martin itself in the particular arrangement recited in claim 14.

D. Group IV – Claims 31, 55 and 79

Regarding claims 31, 55 and 79, Martin does not teach flagging means for flagging the at least one control point on the basis of being active or the instructions for flagging. As previously mentioned, Martin does not even recognize that a control point may be active or non active. Thus, Martin cannot flag a control point based on whether it is active.

The Office Action merely points to its rejection of claim 1 in rejecting this feature. However, none of the sections cited in the rejection of claim 1, nor the rejection itself, address anything remotely resembling flagging a control point on the basis of the control point being active. Martin does not even discuss active and non-active control points.

Examiner's Response to Appellants' Arguments Regarding Group IV

The only response offered by the Examiner with regard to Appellants' argument regarding the claims of Group IV is to "see the argument related to "active" control point above." The argument referred to is merely stating that "flow control" is interpreted to include determining execution paths based on parameters and somehow this suddenly anticipates the specific feature of a flagging means for flagging a control point as active. As discussed above, there is no designation of active or non-active control points in the general concept of "flow control." Moreover, nothing in "flow control" teaches to flag control points using a flagging means to identify control points that are active. Furthermore, the Examiner has not identified anywhere in Martin where a flagging means is specifically taught. Instead, the Examiner relies on generalities regarding alleged "flow control", which is not taught in the Martin reference and is not claimed in the present claims, rather than actually finding the features of Group IV in any reference. Thus, the Examiner has failed to identify any reference that anticipates the features of Group IV, let alone support the allegation that Martin teaches these features.

E. Group V – Claims 32, 43, 45, 56, 67, 69, 80, 91 and 93

With regard to claims 32, 43, 45, 56, 67, 69, 80, 91 and 93, Martin does not teach defining a rule selection algorithm associated with the at least one control point. As previously mentioned above, Martin does not teach any mechanism for selecting rules, let alone a rule selecting algorithm that is associated with a control point. The Office Action alleges that Martin teaches this feature at page 168. However, there is nothing on page 168, or in any other section of Martin, that teaches or even remotely suggests a rule selection algorithm that is associated with a control point. The Martin reference merely teaches that rules may be used to implement event diagram preconditions, post conditions and control conditions. There is no selection mechanism associated with these preconditions, post conditions and control conditions. This is because there is no selection of rules from a set of rules in the teachings of Martin, as previously discussed above.

Examiner's Response to Appellants' Arguments Regarding Group V

In response to this argument, the Examiner merely states "see the argument related to "active" control point above." The argument the Examiner alleged with regard to "active" control points does not address the feature of defining a rule selection algorithm associated with at least one control point. The fact is, the Examiner cannot point to anything in Martin that explicitly teaches a rule selection algorithm and thus, the Examiner tries to rely on a generalization regarding "flow control." Once again, the alleged "flow control" has nothing to do with the features recited in the claims of Group V. The Examiner has not shown where the features of Group V are identically shown in the Martin reference arranged in the manner recited in the pending claims as is required under 35 U.S.C. § 102. Thus, the Examiner has not established a case of anticipation with regard to these claims.

F. Group VI – Claims 33, 44, 57, 68, 81 and 92

With regard to claims 33, 44, 57, 68, 81 and 92, Martin does not teach defining a rule result combination algorithm associated with the at least one control point. The Examiner alleges that a control condition can function as a combining algorithm (Final Office Action, page 6). The Examiner appears to be alleging first, that control conditions are control points and that control conditions are also combining algorithms. While Martin teaches that the preconditions may be implemented as rules, there is no teaching in Martin regarding defining a rule result combination algorithm for combining the results of the rules of a control point. The Martin reference states that the precondition may be implemented as an IF structure but this is a single rule. There is no combination of the results of this rule with any other rule in Martin. Thus, Martin does not teach each and every feature of claims 33, 44, 57, 68, 81 and 92.

G. Group VII – Claims 34, 58 and 82

Regarding claims 34, 58 and 82, Martin does not teach either defining a rule selection algorithm or defining a rule result combination algorithm for at least one control point, as discussed above with regard to claims 56, 57, 80 and 81. Moreover, since Martin does not teach either of these features individually, Martin cannot be found to teach the combination of these two features recited in claims 34, 58 and 82. Thus, Martin does not teach each and every feature of claims 34, 58 and 82.

Examiner's Response to Appellants' Arguments Regarding Group VII

The Examiner's response to these arguments is the same as the arguments addressed above with regard to Group V. Therefore, the Examiner's response is traversed for the same reasons as noted above with regard to Group V and furthermore, the reasons noted above with regard to Group VI.

H. Group VIII – Claim 16

With regard to claim 16, Martin does not teach defining rules to at least one control point on the basis of an object's class name, method name, and position of the at least one control point in the method. The Examiner alleges that Martin teaches these features for the same reasons as noted with regard to claim 1. Thus, for the same reasons as noted above with regard to claim 1, Appellants respectfully submit that Martin does not actually teach this feature.

In addition, Appellants respectfully submit that claim 1 recites associating a set of rules with each control point based on a class of object, name of method and type of control point. Claim 1 does not recite that the rules are associated based on a position of a control point in the method. Martin does not teach this feature and the Examiner has failed to show where this feature may be found anywhere in the Martin reference. Thus, the Examiner has not established a case of anticipation with regard to claim 16.

Examiner's Response to Appellants' Arguments Regarding Group VIII

The Examiner responds to the above arguments with regard to claim 16 by stating "Same arguments as above." The Examiner's "arguments above" do not address associating rules based on a position of a control point in a method. Thus, the Examiner has failed to consider all of the features of the claim and has not established anticipation based on the Martin reference with regard to this feature. Moreover, as discussed above, Martin also does not teach the features of associating rules based on an object's class name, method name and position of the control point in a method.

Thus, Applicants respectfully submit that Martin does not teach each and every feature recited in claim 16 as is required under 35 U.S.C. § 102. At least by virtue of their dependency on claim 16, Martin also does not teach the features recited in dependent claims 17-22. Accordingly, Applicants respectfully request withdrawal of the rejection of claims 16-22 under 35 U.S.C. § 102.

I. Group IX – Claims 17 and 22

In addition to the above with regard to claim 16, Martin also does not teach many of the specific features set forth in dependent claims 17-22. For example, Martin does not teach activating at least one control point having associated rules, as recited in claim 17. As discussed in depth above, Martin does not even recognize the possibility of having active and nonactive control points, let alone activating a control point. Similarly, this lack of teaching in Martin applies to claim 22 which recites "deactivating the at least one control point." Again Martin does not teach active or nonactive control points and thus, cannot teach deactivating a control point.

Examiner's Response to Appellants' Arguments Regarding Group IX

In response to the above arguments, the Examiner merely responds "See arguments as per above." None of the Examiner's arguments identify where in the Martin reference the activation or deactivation of control points is taught. The Examiner appears to be referring to his erroneous interpretation regarding "flow control" which has been addressed in detail above. Once again, Appellants respectfully submit that there is nothing in the general concept of "flow control" that teaches to activate or deactivate control points. Thus, the Examiner has not shown where the Martin reference identically teaches the features of claims 17 and 22 as is required under 35 U.S.C. § 102.

J. Group X – Claim 18

Regarding claim 18, Martin does not teach affecting behavior of the object based on running the rules associated with a control point. As noted above with regard to claim 16, Martin does not teach associating rules with a control point as recited in claim 16. Furthermore, Martin does not teach affecting behavior of an object based on running the rules associated with the control point. The Examiner alleges that this feature is taught on page 381 of Martin but, as with every other allegation made by the Examiner, there simply is nothing on page 381 or any other page of the voluminous Martin reference that teaches affecting the behavior of an object based on running the rules associated with a control point, the rules being associated with the control point

based on an object class, method name, and position in the method.

Examiner's Response to Appellants' Arguments Regarding Group X

The Examiner's response to the arguments regarding claim 18 is to see the argument regarding "active" control points. There is nothing in the Examiner's argument regarding "active" control points that addresses the specific features of claim 18. Thus, the Examiner has failed to show where the Martin reference identically teaches the features of claim 18. Therefore, the Examiner has failed to support his allegation of anticipation of the features of claim 18.

K. Group XI – Claim 19

With regard to claim 19, Martin does not teach that affecting the behavior of the object includes associating different rules to a control point. The Examiner alleges that this feature is anticipated because flow control may be affected by time of day. This has nothing to do with the specific features of claim 19. Nothing in Martin teaches running rules associated with a control point to affect an object wherein the affect is to associate different rules to the control point, as recited in claim 19. The Examiner simply fails to address this specific feature and rather, rests on generalizations that have nothing to do with the specific feature of the claim.

Examiner's Response to Appellants' Arguments Regarding Group XI

The Examiner's response to the arguments regarding claim 19 is to see the argument regarding "active" control points. There is nothing in the Examiner's argument regarding "active" control points that addresses the specific features of claim 19. Thus, the Examiner has failed to show where the Martin reference identically teaches the features of claim 19. Therefore, the Examiner has failed to support his allegation of anticipation of the features of claim 19.

L. Group XII – Claim 20

Regarding claim 20, Martin does not teach that affecting the behavior of an object includes defining another control point. The Examiner again puts forth irrelevant allegations regarding flow control and then alleges that the features of claim 20 are taught by Martin at page 163. Page 163 is equally irrelevant to the claimed feature as the other allegations regarding Martin made by the Examiner. There is nothing on page 163 or any other page of Martin that teaches running rules associated with a control point to affect the behavior of an object where the affect is to define another control point. Page 163 merely illustrates an event diagram for registering students. There is nothing in this event diagram that teaches to run rules of control point to affect the behavior of an object such that the affect includes another control point being defined.

Examiner's Response to Appellants' Arguments Regarding Group XII

The Examiner's response to the arguments regarding claim 20 is to see the argument regarding "active" control points. There is nothing in the Examiner's argument regarding "active" control points that addresses the specific features of claim 20. Thus, the Examiner has failed to show where the Martin reference identically teaches the features of claim 20. Therefore, the Examiner has failed to support his allegation of anticipation of the features of claim 20.

M. Group XIII – Claim 21

Regarding claim 21, Martin does not teach that affecting the behavior of an object includes associating rules to a second control point. The Examiner again alleges that this feature is taught on page 163 of Martin. Nowhere in the event diagram of page 163 of Martin is there any teaching regarding the running of rules associated with a control point to affect the behavior of the object wherein the affect is to associate rules with a second control point.

Examiner's Response to Appellants' Arguments Regarding Group XIII

The Examiner's response to the arguments regarding claim 21 is to see the argument regarding "active" control points. There is nothing in the Examiner's argument regarding "active" control points that addresses the specific features of claim 21. Thus, the Examiner has failed to show where the Martin reference identically teaches the features of claim 21. Therefore, the Examiner has failed to support his allegation of anticipation of the features of claim 21.

N. Group XIV – Claims 23 and 27

With regard to claims 23 and 27, Martin does not teach a method context control point, as previously discussed in detail above with regard to Group II. Specifically, claim 23 recites defining first and second control points of a method, the first and second control points also being a method context control point. Claim 27 recites defining at least one control point, wherein the at least one control point is a method context control point and placing this method context control point in a method after the method logic of the method. Martin does not teach these features of claims 23 and 27.

Examiner's Response to Appellants' Arguments Regarding Group XIV

The Examiner's only response to Appellants' arguments regarding claims 23 and 27 is that "some of these claims were found allowable but not for the limitation argued." Appellants respectfully submit that the Examiner has not allowed either of claims 23 or 27 and the Examiner has not addressed Appellants' arguments. If the Examiner is referring to his indication of persuasive argument on page 42 of the Final Office Action, then the Examiner is again mistaken as to what he has already indicated to be allowable. On page 42 of the Final Office Action, the Examiner indicates that Appellants' arguments (reproduced on page 41) are persuasive. These arguments illustrate that the Martin reference does not teach method context control points. The Examiner has agreed that this argument is persuasive. Claims 23 and 27 recite method context control points and thus, should be allowed for the same reasons as stated with regard to the other claims that recite method context control points.

Furthermore, Martin does not, in fact, actually teach method context control points. As previously argued, a method context control point is a control point that may have a variety of different rules and different types of rules associated with it as the need for these rules change over time (page 11, lines 17-29 of the present specification). The precondition, post-condition and control conditions of Martin are not intended to change over time and are not intended to have different rules and different types of rules associated with them. This is because these conditions are linked to the operation so closely that the conditions cannot be changed without causing the operation to not execute properly. Thus, the precondition, post-condition and control conditions of Martin are not the same as a method context control point and Martin does not teach any other structure that anticipates a method context control point.

Thus, Applicants respectfully submit that Martin does not teach each and every feature recited in claims 23 and 27 as is required under 35 U.S.C. § 102(b). At least by virtue of their dependency on claims 23 and 27, respectively, Martin also does not teach the features recited in dependent claims 24-26, 28 and 31-35. Accordingly, Applicants respectfully request withdrawal of the rejections of claims 23-28 and 31-35 under 35 U.S.C. § 102(b).

O. Group XV – Claims 36-39, 41, 60-63, 65, 84-87 and 89

With regard to claims 36-39, 41, 60-63, 65, 84-87 and 89, Martin does not teach associating a rule with another method within an object class or associating the rule with another object class. As previously mentioned above, Martin only teaches associating rules with preconditions, post conditions and control conditions in an event diagram. There is no teaching in Martin that the same rule may be associated with two different methods in an object class or associating the same rule with two different object classes. While similar rules may be associated with different conditions in an event diagram, there is nothing in Martin that teaches that the same rule may be associated with two methods within an object class or two object classes.

The Examiner alleges that this feature is taught on pages 266-268 of Martin. The Examiner alleges that this section of Martin teaches associating rules with more than one object, reuse and inheritance which can all be read on the claimed feature. However, the general teachings of reuse and inheritance do not in themselves teach the specific features recited in

claims 36-39, 60-65 and 84-89. There is nothing in Martin that teaches or even suggests to one of ordinary skill in the art that the same rule can be associated with a plurality of methods, as recited in claims 36, 60 and 84 or associated with two different object classes. The Examiner is again engaged in reading into general teachings, the specific teachings found only in Appellants' disclosure without any basis for such reading in of teachings in the reference itself.

Thus, Applicants respectfully submit that Martin does not teach each and every feature recited in claims 36, 60 and 84 as is required under 35 U.S.C. § 102(b). At least by virtue of their dependency on claims 36, 60 and 84, Martin also does not teach the features recited in dependent claims 37-39, 41, 61-63, 65, 85-87 and 89. Accordingly, Applicants respectfully request withdrawal of the rejection of claim 36-39, 41, 60-63, 65, 84-87 and 89 under 35 U.S.C. § 102.

Examiner's Response to Appellants' Arguments Regarding Group XV

The Examiner responds to the above arguments by stating, with regard to claims 36, 60 and 84, "the ability to call "helper" methods within an object is considered inherent and present in languages such as ANSI standard C++." This has nothing to do with the actual features of claims 36, 60 and 84. Claim 36, 60 and 84 recite associating the rule with another method within the object class. They do not recite "helper" methods and there is no teaching in Martin regarding "helper" methods associating rules with two methods within an object class. The Examiner has failed to show where the Martin reference explicitly teaches this feature and thus, has not met his burden of establishing a case of anticipation based on the Martin reference.

P. XVI – Claims 42, 47, 66, 71, 90 and 95

Regarding claims 42, 47, 66, 71, 90 and 95, Martin does not teach the feature of determining if a control point is active. This feature has been address above with regard to claim 1 and thus, claims 42 and 47 define over Martin for similar reasons as noted above with regard to this feature in claim 1. Nothing in Martin teaches that control points may be active or inactive and thus, there is no reason why Martin would teach determining if a control point is active. To the contrary, if a condition exists in the diagram of Martin, it must be active – it either exists or it

does not. There is no ability for a condition to exist and it not be active in Martin.

Thus, Appellants respectfully submit that Martin does not teach each and every feature recited in claims 42, 47, 66, 71, 90 and 95 as is required under 35 U.S.C. § 102. At least by virtue of their dependency on claims 42, 47, 66, 71, 90 and 95, respectively, Martin does not teach the features recited in claims 43-44, 67-68, 91-92. Accordingly, Appellants respectfully request withdrawal of the rejection of claims 42-44, 47, 66-68, 71, 90-92 and 95 under 35 U.S.C. § 102.

Examiner's Response to Appellants' Arguments Regarding Group XVI

The Examiner does not offer any new rebuttal to Appellants arguments with regard to claims 42-44, 47, 66-68, 71, 90-92 and 95. The Examiner's response is "It appears we have reached a level which dependent claims are relying on prior non persuasive arguments." Thus, the Examiner has not provided any support for the allegation that the features of these claims are anticipated by Martin and merely relies on the erroneous interpretation of Martin previously addressed. Therefore, Appellants respectfully submit that Martin does not teach determining if a control point is active as discussed above and thus, Martin does not anticipate claims 42-44, 47, 66-68, 71, 90-92 and 95.

Q. Group XVII – Claims 73 and 97

With regard to claims 73 and 97, Martin does not teach controlling a method by exiting the method. As noted above, the operation in Martin may be implemented as a method. For the method to be invoked, the precondition must be satisfied and the control condition must be satisfied. Thus, the only options are that the operation is either executed because both the precondition and the control conditions are satisfied or the operation is not executed because one or both of the precondition and control condition are not satisfied. There is no possibility of "exiting" the operation based on the precondition and the control condition because the operation is not entered until after the precondition and control condition operate. Thus, Martin does not teach controlling the method by exiting the method as described in claims 73 and 97.

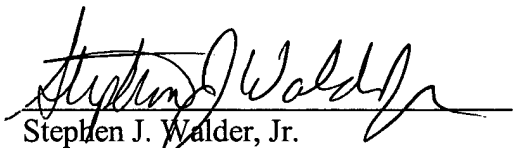
Examiner's Response to Appellants' Arguments Regarding Group VIII

The Examiner responds to the above arguments by stating that "exiting operations are inherent in programming." While exiting a method may be known in the art in general, there is no teaching in the prior art, and no teaching has been identified in the Martin reference, that controlling a method on the basis of results obtained from running at least one rule associated with a method context control point includes exiting the method. The Examiner has failed to examine this feature and has merely excised the function of "exiting" from the context of claims 73 and 97. Thus, while "exiting" in general may be known, exiting in the manner recited in claims 73 and 97 was not known prior to Appellants' invention. Thus, the Examiner has once again failed to establish a case of anticipation with regard to claims 73 and 97.

Conclusion

In view of the above, Appellants respectfully submit that Martin does not teach each and every feature of claims 1, 12-28, 31-39, 41-47, 51-52, 55-63, 65-69, 71-76, 79-87, 89-93 and 95-98 as is required under 35 U.S.C. § 102. Therefore, Appellants respectfully request that the Board of Patent Appeals and Interferences overturn the rejection of these claims.

Respectfully submitted,



Stephen J. Walder, Jr.
Reg. No. 41,534
Carstens, Yee & Cahoon, LLP
PO Box 802334
Dallas, TX 75380
(972) 367-2001

APPENDIX OF CLAIMS

The text of the claims involved in the appeal are:

1. A computer implemented process for applying a set of rules, the process comprising:

(a) placing a pre-method control point before logic of a method and post-method control point after the logic of the method;

(b) associating a set of rules with each control point based on a class of object in which the method resides, name of the method, and type of control point, whether the pre-method control point or the post-method control point;

(c) invoking the method, wherein encountering each control point during the execution of the method comprises:

(i) determining if the encountered control point is active;

(ii) on the basis of an active control point:

1) selecting rules based on a set of rules associated with the active control point associated in step (b);

2) running the selected rules;

3) obtaining results from running the rules; and

4) combining the results using a combining algorithm specified by the control point.

12. A computer implemented process for applying a set of rules, comprising:

(a) invoking a method in an object;

(b) encountering an active control point during the invocation of the method, wherein

the active control point is an active method context control point;

(c) selecting rules associated with the method of the object at the control point;

(d) invoking the rules; and

(e) combining results from invoking the rules.

13. The process of claim 12, wherein the rules perform a variety of actions conditioned by the fact that rules may be associated with particular, regularly occurring points in the object model.

14. The process of claim 12, wherein the rules perform at least one function which varies over time.

15. A process of claim 12, wherein a control point occurs just before logic of the method begins, just after the logic of the method completes, or at both just before logic of the method begins and just after the logic of the method completes.

16. A computer implemented process for applying a set of rules comprising:

(a) defining an object;

(b) defining at least one method in the object;

(c) defining at least one control point in the at least one method;

(d) defining rules to the at least one control point on basis the object's class name, method, name, and position of the at least one control point in the method.

17. In the process of claim 16, further comprising the step of activating at least one control point having associated rules.

18. The process of claim 16 further comprises:

(e) encountering a first control point;

(f) running the rules associated with the first control point; and

(g) affecting behavior of the object base on running the rules associated with the first control point.

19. In the process of claim 18, the step of affecting the behavior of the object further comprises:

(h) associating different rules to a control point.

20. In the process of claim 18, the step of affecting the behavior of the object further comprises:

(h) defining another control point.

21. In the process of claim 18, the step of modifying the object further comprises:

(h) associating rules to a second control point.

22. In the process of claim 16, further comprising a step of deactivating the at least one control point.

23. A computer implemented process for applying a set of rules, comprising:

(a) defining an object;

(b) defining a method in the object;

(c) defining a first control point of the method, the first control point being a method context control point;

(d) determining rules associated with the first control point;

(e) defining a second control point of the method, the second control point also being a method context control point; and

(f) determining rules associated with the second control point.

24. A computer implemented process as in claim 23 further comprising:

(g) separately selecting, running and combining the results of rules determined to be associated with either control point.

25. In the process of claim 23 wherein the first control point is a pre-method trigger point.

26. In the process of claim 23 wherein the second control point is a post-method trigger point.

27. A computer implemented process for defining an object comprising:

defining an object;

defining a method in the object by:

defining method logic;

placing the method logic in the method;

defining at least one control point, wherein the at least one control point is a method context control point; and

placing the at least one control point in the method wherein the method logic is continuous, wherein the step of placing the at least one control point further comprises placing the at least one control in the method after the method logic.

28. A computer implemented process for defining an object as in claim 27, wherein the step of placing the at least one control point further comprises placing the at least one control in the method before the method logic.

31. A computer implemented process for defining an object as in claim 27, further comprises:

flagging the at least one control point on the basis of being active.

32. A computer implemented process for defining an object as in claim 27, wherein the step of defining the at least one control point further comprising:

defining a rule selection algorithm associated with the at least one control point.

33. A computer implemented process for defining an object as in claim 27, wherein the step of defining the at least one control point further comprising:

defining a rule result combination algorithm associated with the at least one control point.

34. A computer implemented process for defining an object as in claim 27, wherein the step of defining the at least one control point further comprises:

defining a rule selection algorithm for the at least one control point; and

defining a rule result combination algorithm for the at least one control point.

35. A computer implemented process for defining an object as in claim 27, further comprising:

associating at least one rule with the at least one control point.

36. A computer implemented process for defining a rule comprising:

creating the rule;

associating the rule with an object class;

associating the rule with a method within the object class;

associating the rule with an occurrence of a control point within the method; and

associating the rule with another method within the object class.

37. A computer implemented process for defining a rule as in claim 36 wherein the occurrence of the control point within the method being before method logic.

38. A computer implemented process for defining a rule as in claim 36 wherein the occurrence of control point within the method being after method logic.

39. A computer implemented process for defining a rule as in claim 36, further comprising:

associating the rule with another object class.

41. A computer implemented process for defining a rule as in claim 36, further comprising:

associating the rule with another control point within the method of the object class.

42. A computer implemented process for applying a set of rules, comprising:

selecting an object class;

selecting a method within the object class;

invoking the method;

processing rules associated with the method comprising:

encountering a control point associated with the method;

determining if the control point is active; and

finding at least one rule associated with an active control point.

43. A computer implemented process for applying a set of rules as in claim 42, wherein the step of finding at least one rule further comprises:

accessing a selecting algorithm associated with the active control point; and

selecting at least one rule using the selecting algorithm.

44. A computer implemented process for applying a set of rules as in claim 42, where in the step of processing rules further comprises:

running the at least one rule;

determining results from running the at least one rule;

accessing a combining algorithm associated with the control point; and

combining the results using the combining algorithm.

45. A computer implemented process for applying a set of rules, comprising:

selecting an object class;

selecting a method within the object class;

invoking the method;

processing rules comprising:

encountering a control point;

accessing a selecting algorithm associated with the control point; and

selecting at least one rule using the selecting algorithm.

46. A computer implemented process for applying a set of rules, comprising:

selecting an object class;

selecting a method within the object class;

invoking the method;

processing rules comprising:

encountering a method context control point;

finding at least one rule associated with the method context control point;

running the at least one rule;

determining results on the basis of running the at least one rule;

accessing a combining algorithm associated with the method context control point; and

combining the results using the combining algorithm.

47. A computer implemented process for applying a set of rules, comprising:

selecting an object class;

selecting a method within the object class;

invoking the method;

processing rules comprising:

encountering a first control point associated with the method;

determining if the first control point is active;

executing method logic of the method;

encountering a second control point associated with the method;

determining if the second control point is active; and

finding a set of rules associated with one of the first control point and the second control point, wherein the set of rules contains not less than zero rules.

51. A data processing system for defining an object comprising:

defining means for defining an object;

defining means for defining a method in the object by:

defining means for defining method logic;

placing means for placing the method logic in the method;

defining means for defining at least one control point, wherein the at least one control point is a method context control point; and

placing means for placing the at least one control point in the method wherein the method logic is continuous, wherein the step of placing the at least one control point further comprises placing means for placing the at least one control in the method after the method logic.

52. A data processing system for defining an object as in claim 51, wherein the step of placing the at least one control point further comprises placing means for placing the at least one control in the method before the method logic.

55. A data processing system for defining an object as in claim 51, further comprises:

flagging means for flagging the at least one control point on the basis of being active.

56. A data processing system for defining an object as in claim 51, wherein the step of defining the at least one control point further comprising:

defining means for defining a rule selection algorithm associated with the at least one control point.

57. A data processing system for defining an object as in claim 51, wherein the step of defining the at least one control point further comprising:

defining means for defining a rule result combination algorithm associated with the at least one control point.

58. A data processing system for defining an object as in claim 51, wherein the step of defining the at least one control point further comprises:

defining means for defining a rule selection algorithm for the at least one control point; and

defining a rule result combination algorithm for the at least one control point.

59. A data processing system for defining an object as in claim 51, further comprising:

associating means for associating at least one rule with the at least one control point.

60. A data processing system for defining a rule comprising:

creating means for creating the rule;

associating means for associating the rule with an object class;

associating means for associating the rule with a method within the object class;

associating means for associating the rule with an occurrence of a control point within the method; and

associating means for associating the rule with another method within the object class.

61. A data processing system for defining a rule as in claim 60 wherein the occurrence of the control point within the method being before method logic.

62. A data processing system for defining a rule as in claim 60 wherein the occurrence of control point within the method being after method logic.

63. A data processing system for defining a rule as in claim 60, further comprising:

associating means for associating the rule with another object class.

65. A data processing system for defining a rule as in claim 60, further comprising:

associating means for associating the rule with another control point within the method of the object class.

66. A data processing system for applying a set of rules, comprising:

selecting means for selecting an object class;

selecting means for selecting a method within the object class;

invoking means for invoking the method;

processing means for processing rules associated with the method comprising:

encountering means for encountering a control point associated with the method;

determining means for determining if the control point is active; and

finding means for finding at least one rule associated with an active control point.

67. A data processing system for applying a set of rules as in claim 66, wherein the step of finding at least one rule further comprises:

accessing means for accessing a selecting algorithm associated with the active control point; and

selecting means for selecting at least one rule using the selecting algorithm.

68. A data processing system for applying a set of rules as in claim 66, where in the step of processing rules further comprises:

running means for running the at least one rule;

determining means for determining results from running the at least one rule;

accessing means for accessing a combining algorithm associated with the control point; and

combining means for combining the results using the combining algorithm.

69. A data processing system for applying a set of rules, comprising:

selecting means for selecting an object class;

selecting means for selecting a method within the object class;

invoking means for invoking the method;

processing means for processing rules comprising:

encountering means for encountering a control point;

accessing means for accessing a selecting algorithm associated with the control point; and

selecting means for selecting at least one rule using the selecting algorithm.

71. A data processing system for applying a set of rules, comprising:

selecting means for selecting an object class;

selecting means for selecting a method within the object class;

invoking means for invoking the method;

processing means for processing rules comprising:

encountering means for encountering a first control point associated with the method;

determining means for determining if the first control point is active;

executing means for executing method logic of the method;

encountering means for encountering a second control point associated with the method;

determining means for determining if the second control point is active;

finding means for finding a set of rules associated with one of the first control point and the second control point, wherein the set of rules contains not less than zero rules.

72. A data processing system for applying a set of rules, comprising:

selecting means for selecting an object class;

selecting means for selecting a method within the object class;

invoking means for invoking the method;

processing means for processing rules comprising:

encountering means for encountering a control point associated with the method, the control point being a method context control point;

finding means for finding at least one rule associated with the control point prior to executing method logic of the method;

running means for running the at least one rule;

obtaining means for obtaining results on the basis of running the at least one rule; and

controlling means for controlling the method on the basis of the results.

73. A data processing system for applying a set of rules as in claim 72, wherein the step of controlling the method comprises:

exiting means for exiting the method.

74. A data processing system for applying a set of rules as in claim 72, wherein the step of controlling the method comprises:

executing means for executing method logic of the method.

75. A computer program product embodied on a computer readable medium containing instructions for a computer implemented process for defining an object, the instruction comprising:

instructions for defining an object;

instructions for defining a method in the object by:

instructions for defining method logic;

instructions for placing the method logic in the method;

instructions for defining at least one control point, wherein the at least one control point is a method context control point; and

instructions for placing the at least one control point in the method wherein the method logic is continuous, wherein the step of placing the at least one control point further comprises placing the at least one control in the method after the method logic.

76. A computer program product for defining an object as in claim 75, wherein the instruction of placing the at least one control point further comprises placing the at least one control point in the method before the method logic.

79. A computer program product for defining an object as in claim 75, further comprises:

instructions for flagging the at least one control point on the basis of being active.

80. A computer program product for defining an object as in claim 75, wherein the instruction of defining the at least one control point further comprises:

instructions for defining a rule selection algorithm associated with the at least one control point.

81. A computer program product for defining an object as in claim 75, wherein the instruction of defining the at least one control point further comprises:

instructions for defining a rule result combination algorithm associated with the at least one control point.

82. A computer program product for defining an object as in claim 75, wherein the step of defining the at least one control point further comprises:

instructions for defining a rule selection algorithm for the at least one control point; and

instructions for defining a rule result combination algorithm for the at least one control point.

83. A computer program product for defining an object as in claim 75, further comprising:

instructions for associating at least one rule with the at least one control point.

84. A computer program product embodied on a computer readable medium containing instructions for a computer implemented process for defining a rule, the instruction comprising:

instructions for creating the rule;

instructions for associating the rule with an object class;

instructions for associating the rule with a method within the object class;

instructions for associating the rule with an occurrence of a control point within the method; and

instructions for associating the rule with another method within the object class.

85. A computer program product for defining a rule as in claim 84 wherein the occurrence of the control point within the method being before method logic.

86. A computer program product for defining a rule as in claim 84 wherein the occurrence of control point within the method being after method logic.

87. A computer program product for defining a rule as in claim 84, further comprising:

instructions for associating the rule with another object class.

89. A computer implemented process for defining a rule as in claim 84, further comprising:

instructions for associating the rule with another control point within the method of the object class.

90. A computer program product embodied on a computer readable medium containing instructions for a computer implemented process for applying a set of rules, the instruction comprising:

instructions for selecting an object class;

instructions for selecting a method within the object class;

instructions for invoking the method;

instructions for processing rules associated with the method comprising:

instructions for encountering a control point associated with the method;

instructions for determining if the control point is active; and

instructions for finding at least one rule associated with an active control point.

91. A computer program product for applying a set of rules as in claim 90, wherein the step of finding at least one rule further comprises:

instructions for accessing a selecting algorithm associated with the active control point; and

instructions for selecting at least one rule using the selecting algorithm.

92. A computer program product for applying a set of rules as in claim 90, where in the step of processing rules further comprises:

instructions for running the at least one rule;

instructions for determining results from running the at least one rule;

instructions for accessing a combining algorithm associated with the control point; and

instructions for combining the results using the combining algorithm.

93. A computer program product embodied on a computer readable medium containing instructions for a computer implemented process for applying a set of rules, the instruction comprising:

instructions for selecting an object class;

instructions for selecting a method within the object class;

instructions for invoking the method;

instructions for processing rules comprising:

instructions for encountering a control point;

instructions for accessing a selecting algorithm associated with the control point; and

instructions for selecting at least one rule using the selecting algorithm.

95. A computer program product embodied on a computer readable medium containing instructions for a computer implemented process for applying a set of rules, the instruction comprising:

instructions for selecting an object class;

instructions for selecting a method within the object class;

instructions for invoking the method;

instructions for processing rules comprising:

instructions for encountering a first control point associated with the method;

instructions for determining if the first control point is active;

instructions for executing method logic of the method;

instructions for encountering a second control point associated with the method;

instructions for determining if the second control point is active;

instructions for finding a set of rules associated with one of the first control point and the second control point, wherein the set of rules contains not less than zero rules.

96. A computer program product embodied on a computer readable medium containing instructions for a computer implemented process for applying a set of rules, the instruction comprising:

instructions for selecting an object class;

instructions for selecting a method within the object class;

instructions for invoking the method;

processing rules comprising:

instructions for encountering a control point associated with the method, the control point being a method context control point;

instructions for finding at least one rule associated with the control point prior to executing method logic of the method;

instructions for running the at least one rule;

instructions for obtaining results on the basis of running the at least one rule; and

instructions for controlling the method on the basis of the results.

97. A computer program product for applying a set of rules as in claim 96, wherein the step of controlling the method comprises:

instructions for exiting the method.

98. A computer program product for applying a set of rules as in claim 96, wherein the step of controlling the method comprises:

instructions for executing method logic of the method.